Implementation of the Fog Computing Approach into Geospatial Web Services

Evgeny Panidi

Saint Petersburg State University, St. Petersburg, Russia; panidi@ya.ru, e.panidi@spbu.ru

Abstract: A number of publications on the information technologies subject in last few years highlights a slow shift from a centralized Cloud-based architecture of information systems (large-scale data centers/thin clients) to decentralized architecture (large-scale data centers/partial storage and computation on client devices). This approach is denoted as Fog Computing or Fog Networking. This shot paper highlights need and possibility of involvement of the Fog Computing approach when developing geospatial Web services.

Keywords: Geospatial Web Services, Geospatial Fog Computing, Geospatial Data Interoperability

1. Introduction

Fog Computing paradigm was proposed in 2012 (Bonomi et al. 2012). It assumes incorporation of client devices into the Cloud. Three key features characterize this incorporation. First, storing of the significant data amount on client devices. Second, providing of the significant amount of communications directly between client devices. Third, performing of the significant amount of data management procedures on client devices (OpenFog Consortium Architecture Working Group 2016, 2017). As a result, the Cloud bounds are blurred, and computational/storage infrastructure becomes composed of Cloud core and connected Fog nodes (incorporated client devices).

In the domain of geospatial standardization and GIS design, the issues of partial moving of the functionality to client devices that interact with the geospatial Cloud remain almost unexplored. Only few publications reviewed case studies dealt with the client-side or decentralized computations in the Cloud-based or distributed GISs (Hakestad 2006; Coene at al. 2007; Keens (ed.) 2007; Panidi et al. 2015). On the other hand, geospatial technologies are developed in a way of general information technologies evolution, and the problem of accumulation of the large geospatial data volumes in real-time mode is relevant already. It is supported also, by well-established understanding of the geospatial component as an essential part of data generated and accumulated by humans (i.e., wide use of geotags, geospatial search embedded into the Web search portals, etc.).

Additionally, evolution of approaches to the collection of geospatial data also implies the necessity of technology development for pre-treatment and on-the-flight data analysis directly on the nodes of the geospatial computer networks (data collecting nodes or data aggregating nodes), with the subsequent transfer of some selected data to the Cloud for storing and retrospective analysis. This is exactly the same technological chain that is assumed in Fog networks.

Some research and developments are provided already in this way. In particular, the Open Geospatial Consortium Sensor Web Enablement framework (OGC SWE) is developed (Open Geospatial Consortium 2007, 2008). It incorporates a couple of open standards, which are regulate design, implementation and management of networks of the geospatial smart sensors.

2. Experience Overview

Development of interfaces for Fog operation and interaction is one of the key research issues (Chiang and Zhang 2015). At the same time, it is needed to ensure interoperability with current Cloud-based systems and services, while great number of standardized (compliant to the OGC and other standards) Cloud-based Web services are in use already.

Our research team has realized a series of experimental projects to provide opportunities for processing and storage of geospatial data on the client side, while creating and using geospatial Web services. In result of these projects, three prototypes of software servers were implemented.

Development of the extension for the OGC Web Processing Service standard (Open Geospatial Consortium 2015) was aimed on support of transmitting of the service executable files onto the client and executing geospatial data processing on the client side. The extension of the standard in this case assumes implementation of additional type of requests to the WPS server. This additional request allows to download executable files for client-side processing. All other requests are processed and responded accordingly to the native WPS standard. We denoted this type of WPSs as Hybrid Geoprocessing Web Services (HGWSs). The prototype of HGWS server was implemented for testing of the proposed approach (Kazakov et al. 2015, Panidi et al. 2015).

Two other projects were aimed on implementation of the lightweight outside-the-Cloud portable software servers, which implemented OGC Web Coverage Service standard (Open Geospatial Consortium 2010). Prerequisite of such developments was the need of low-cost Web publication of small-size raster coverages produced in result of different geospatial researches. At the time of development (and currently also) the only way of WCS publication was the use of dedicated server with specialized software. However, the use of dedicated server is inapplicable when it is needed to publish small scientific raster dataset in the form of standardized Web service (WCS). Such a publication is demanded in small-scale scientific projects for example.

Development of the prototypes of portable WCS servers was provided in two ways. First prototype was realized as traditional Common Gateway Interface (CGI) application, which is compatible with almost any Web server and with many virtual shared Web hosting platforms. The prototype does not require installation on the operating system and allows deployment on ordinary (virtual shared) hosting, or on the client's operating system (Panidi et al. 2016). In the second case, this portable WCS server can be deployed on the Fog nodes.

Development of the Node.js-based WCS server was provided as alternative to the CGI WCS server to allow the use of Node.js, which is popular server framework. This WCS server can be deployed also on a virtual shared Node.js hosting or on the client's operating system (Panidi et al. 2016).

Main idea of the abovementioned experimental projects was consist in that the software, which is used as the platform for geospatial Web services, have to be compact and portable, and should allow deployment right here and right now, without preparation of a specialized infrastructure (specialized Cloud or dedicated servers).

3. Conclusions

In our studies, we have completed testing and performance evaluation for the implemented prototypes of the server software for client-side geospatial data processing in distributed computer systems, and for out-of-the-Cloud publication of small-scale geospatial data services (WCSs). These prototypes are applicable for publishing on unspecialized infrastructure and on the Fog nodes.

Preliminary analysis of results allows us to conclude the effectiveness of selected approach that consists in extending of the currently used standards to ensure interoperability with Fog architecture, and in development of server software in the portable manner. Currently, this approach is applicable in the case of small- and medium-scale projects concerned with publication of geospatial data and processing software. However, it is compatible with the Fog approach and can be used as a basis for design of more complex geospatial Fog-based systems.

Most significant aims for further research in the domain of geospatial Fog are the development of Fog interfaces and design of interoperability decisions.

References

Bonomi F., Milito R., Zhu J., Addepalli S. (2012). Fog Computing and Its Role in the Internet of Things. Proceedings of the first edition of the MCC workshop on Mobile cloud computing, 13–16. doi:10.1145/2342509.2342512

Chiang M., Zhang T. (2016). Fog and IoT: An Overview of Research Opportunities. *IEEE Internet of Things Journal*, 3(6), 854–864. doi:10.1109/JIOT.2016.2584538

Coene Y., Marchetti P.G., Smolders S. (2007). Architecture and Standards for a Distributed Digital Library of Geospatial Services. Third Italian Research Conference on Digital Library Systems (IRCDL 2007) Padova 29–30 January 2007 Proceedings, 52–60.

Hakestad O. (2006). Fleet Management Lite Server and Web Client. Ostfold University College, Mobile Applications Group, Halden, Norway, 62 p.

Kazakov E., Terekhov A., Kapralov E., Panidi E. (2015). WPS-based technology for client-side remote sensing data processing. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XL-7/W3, 643–649. doi:10.5194/isprsarchives-XL-7-W3-643-2015

Keens S. (ed.) (2007). *Discussions, findings, and use of WPS in OWS-4*. OGC Discussion Paper. OGC 06-182r1. Version 0.9.1, 2007-05-10, 57 p.

Open Geospatial Consortium. (2008). *OGC Sensor Web Enablement Architecture*. Best Practice document 06-021r4, Version: 0.4.0, 72 p.

Open Geospatial Consortium. (2007). *OGC Sensor Web Enablement: Overview and High Level Architecture*. OpenGIS White Paper OGC 07-165, Version: 3, 14 p.

Open Geospatial Consortium. (2010). *OGC WCS 2.0 Interface Standard - Core*. OGC 09-110r3, Version 2.0.0, 54 p.

Open Geospatial Consortium. (2015). *OGC WPS 2.0 Interface Standard*. *OGC Implementation Standard*. OGC 14-065, Version 2.0, 133 p.

OpenFog Consortium Architecture Working Group. (2016). *OpenFog Architecture Overview*. White Paper OPFWP001.0216, 35 p.

OpenFog Consortium Architecture Working Group. (2017). *OpenFog Reference Architecture for Fog Computing*. OPFRA001.020817, 162 p.

Panidi E., Kazakov E., Kapralov E., Terekhov A. (2015). Hybrid geoprocessing Web services. SGEM2015 Conference Proceedings, 2(1), 669–676. doi:10.5593/SGEM2015/B21/S8.084

Panidi E., Terekhov A., Mukhamedzyanov M. (2016). Implementation Ways for Web Coverage Service Standard. 2nd Internatonal Scientific Conference GEOBALCANICA 2016 Proceedings, 3–10. doi:10.18509/GBP.2016.01